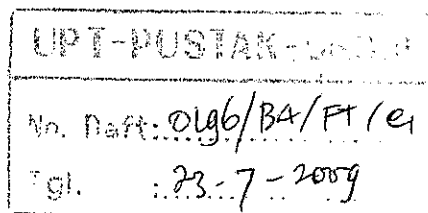


**BUKU AJAR**



**PEMROGRAMAN KOMPUTER  
TEKNIK GEODESI**

**OLEH  
MOEHAMMAD AWALUDDIN, ST**



**PROGRAM STUDI TEKNIK GEODESI  
FAKULTAS TEKNIK  
UNIVERSITAS DIPONEGORO  
2007**

**BAGIAN I**  
**ALGORITMA**

# **ALGORITMA**

Menggunakan algoritma untuk  
menyelesaikan suatu permasalahan dan  
membuat definisi algoritma

# **ALGORITMA**



**ALGORISM / ALGORITHM**



**AL-KHUWARIZMI**

## **Algoritma Memutar Kaset**

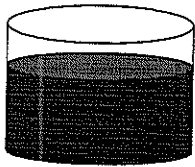
1. Pastikan tape recorder dalam posisi ON
2. Masukkan kaset ke dalam tape recorder
3. Tekan tombol PLAY

## **Buat Algoritma Kasus Berikut**

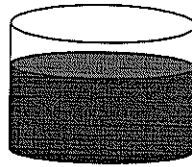
Misalkan terdapat dua buah ember, masing-masing mempunyai volume 5 l dan 3 l. Tuliskan algoritma untuk memperoleh air sebanyak 1 liter dengan hanya menggunakan kedua buah ember tersebut

## Algoritma Tukar Isi

Jika ada 2 buah ember A dan B. Ember A berisi air yang berwarna merah, sedangkan ember B berisi air berwarna biru. Bagaimana algoritma untuk mempertukarkan isi kedua ember tersebut, sehingga Ember A berisi air yang berwarna biru, sedangkan ember B berisi air berwarna merah



A



B

## Algoritma Euclidean

Proses untuk Menemukan Bilangan Pembagi  
Bersama Terbesar

## Algoritma Euclidean

Diberikan dua buah bilangan bulat tak negatif  $m$  dan  $n$  ( $m \geq n$ ). Algoritma euclidean untuk mencari bilangan bulat positif pembagi bersama terbesar (gcd) dari kedua bilangan tersebut adalah:

1. Jika  $n = 0$ , maka  
gcd =  $m$ , stop  
tetapi jika  $n \neq 0$  lanjut ke langkah 1.2
2. Bagilah  $m$  dengan  $n$  dan misalkan  $r$  adalah sisanya.
3. Ganti nilai  $m$  dengan  $n$  dan nilai  $n$  dengan  $r$ , lalu ulangi dari langkah 1.1

**KASUS 1**

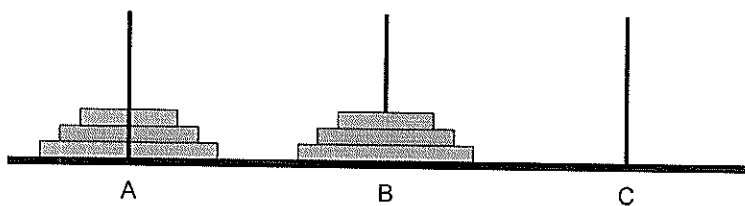
**KASUS 2**

**KASUS 3**

**KASUS 4**

### **KASUS 1**

Tiga buah cakram yang masing-masing berdiameter berbeda mempunyai lubang di titik pusatnya. Ketiga cakram tersebut dimasukkan ke sebuah batang besi A sedemikian hingga cakram yang berdiameter lebih besar selalu terletak di bawah yang kecil. Tulislah algoritma untuk memindahkan seluruh cakram tersebut ke batang besi B; dengan syarat setiap kali hanya satu cakram yang boleh dipindahkan dan setiap perpindahan tidak boleh ada cakram yang lebih besar berada di atas yang kecil. Batang besi C dapat dipakai sebagai bantuan



## **KASUS 2**

Tiga pasang suami istri yang sedang menempuh perjalanan sampai ke sebuah sungai. Mereka menemukan sebuah perahu kecil yang hanya bisa membawa tidak lebih dari dua orang setiap kali menyeberang. Penyeberangan sangat dirumitkan oleh kenyataan bahwa para suami sangat pencemburu dan tidak mau meninggalkan istri mereka jika ada lelaki lain. Tulislah algoritma untuk menunjukkan bagaimana penyeberangan tersebut dapat dilakukan.

Coba Definisikan Arti Algoritma



## **Struktur Program / Algoritma**

1. Struktur Runtunan / Urut
2. Struktur Pemilihan / Keputusan
3. Struktur Pengulangan

## **STRUKTUR URUT**

Setiap baris program akan dikerjakan  
secara urut dari atas ke bawah

## **Algoritma Tukar Isi**

1. Tuangkan air dari ember A ke ember C
2. Tuangkan air dari ember B ke ember A
3. Tuangkan air dari ember C ke ember A

## **Algoritma Hitung Luas Persegi Panjang**

1. Masukkan data panjang = p
2. Masukkan data lebar = l
3. Hitung luas =  $p * l$
4. Tulis Luas

## **Buat Algoritma Berikut**

Diketahui data suhu dalam celcius, ubah data suhu tersebut dalam fahrenheit dan reamur

Diketahui data koordinat A & B, hitung jarak A ke B

Diketahui data pengukuran BA, BB, BT , ti (tinggi alat) dan V (sudut vertikal), hitung jarak & beda tinggi

## **STRUKTUR PEMILIHAN / KEPUTUSAN**

✓ Tidak setiap baris program dikerjakan

baris program dikerjakan jika memenuhi suatu kondisi yang disyaratkan

## **BENTUK STRUKTUR PEMILIHAN / KEPUTUSAN**

```
if kondisi then  
    aksi 1  
else  
    aksi 2
```

### **Algoritma menentukan nilai terbesar dari 2 bilangan x dan y ( $x \neq y$ )**

- 1.1. Masukkan data x dan y
- 1.2. if  $x > y$  then  
 tulis x sebagai bilangan terbesar  
else  
 tulis y sebagai bilangan terbesar

## **Buat Algoritma Berikut**

Menentukan suatu bilangan positif genap atau ganjil

Menentukan bilangan terbesar dari 3 bilangan x, y & z

## **STRUKTUR PENGULANGAN**

Suatu baris program dapat dikerjakan  
secara berulang-ulang

### **Algoritma Hitung Luas 100 lingkaran**

1. Masukkan data jari-jari lingkaran 1 ( $r_1$ )
  2. Hitung Luas Lingkaran 1 =  $\pi * r_1^2$
  3. Tulis Luas Lingkaran 1
  4. Masukkan data jari-jari lingkaran 2 ( $r_2$ )
  5. Hitung Luas Lingkaran 2 =  $\pi * r_2^2$
  6. Tulis Luas Lingkaran 2
- 
298. Masukkan data jari-jari lingkaran 100 ( $r_{100}$ )
  299. Hitung Luas Lingkaran 100 =  $\pi * r_{100}^2$
  300. Tulis Luas Lingkaran 100

### **Algoritma Hitung Luas 100 lingkaran**

1. For i dari 1 sampai 100
2. Masukkan data jari-jari lingkaran ke-i ( $r_i$ )
3. Hitung Luas Lingkaran ke-i =  $\pi * r_i^2$
4. Tulis Luas Lingkaran ke-i
5. Next i

## **STRATEGI PERANCANGAN PUNCAK - TURUN**

- ✓ Algoritma secara global
- ✓ Membuat lebih rinci dari setiap tahapan

### **Algoritma Sortir Ascending / Membesar (Global)**

1. Cari nilai terbesar di antara N buah elemen data
2. Tempatkan nilai terbesar tersebut pada posisi yang tepat
3. Ulangi dari langkah 1 untuk N-1 buah data yang lain

## **Algoritma Sortir Ascending / Membesar (Rinci)**

- 1.1. Asumsikan data ke-1 sebagai data terbesar,  
Maks =  $a(1)$
- 1.2. While belum mencapai data ke-N do  
    tinjau data berikutnya  
    if data berikutnya lebih besar dari maks then  
        ganti maks dengan data tersebut
- 2.1. Masukkan data ke-N di dalam C (Temporary)
- 2.2. Masukkan maks ke data ke - N
- 2.3. Masukkan ke dalam tempat tempat maks yg lama
- 3.1. Kurangi N dengan 1

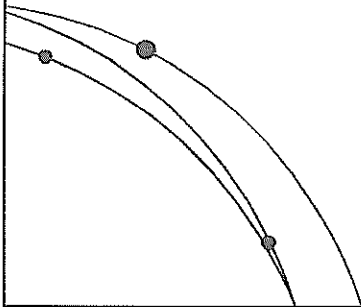
## **Algoritma Sortir**

```
For i dari 1 sampai N-1
  For j=i+1 sampai N
    Bandingkan  $a(i)$  dengan  $a(j)$ 
    If  $a(i) > a(j)$  then lakukan tukar tempat
  Next j
Next i
```



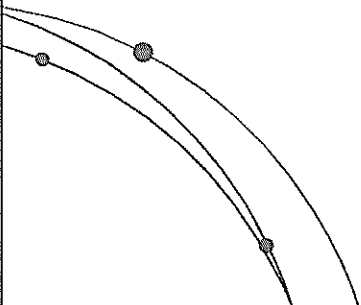
## PENYAJIAN ALGORITMA

Tulisan  
Gambar



## FLOW CHART

Bagan yang menggambarkan urutan instruksi untuk suatu proses data dan juga menggambarkan hubungan antar suatu proses dengan proses yang lainnya dengan menggunakan symbol – symbol tertentu



**BAGIAN II**

**PEMROGRAMAN BASIC**

# **PEMROGRAMAN KOMPUTER**

- ✓ Bahasa Pemrograman Komputer → Basic, Visual Basic
- ✓ Pembuatan program hitungan untuk pekerjaan di bidang survey pemetaan dan geodesi

## **Bahasa Komputer**

1. Bahasa Mesin → Dapat berinteraksi secara langsung dengan komputer → sulit dipahami oleh orang awam
2. Bahasa Awam → berinteraksi dengan komputer melalui bahasa perantara atau compiler → mudah dipahami oleh orang awam

Program → Kumpulan beberapa statemen yang harus disusun dengan dengan urutan yang nalar, untuk mendapatkan informasi yang benar dari serangkaian data.

## **Unsur – unsur dalam Program**

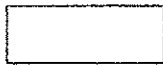
1. Input → data
2. Processing → formula / rumus
3. Output → informasi

Flowchart → Bagan yang menggambarkan urutan instruksi untuk suatu proses data dan juga menggambarkan hubungan antar suatu proses dengan proses yang lainnya dengan menggunakan symbol – symbol tertentu.

## Simbol-simbol flowchart :

### 1. Proses

Simbol :



Menggambarkan instruksi atau proses pengolahan data.

### 2. Operasi masukan-keluaran

Simbol :



Menunjukkan operasi yang dilakukan untuk membaca data dan menulis hasil.

### 3. Penghubung

Dalam satu halaman :

Simbol :



Penghubung antar halaman :

Simbol :




### 4. Tanda panah

Menunjukkan arah aliran proses.

Ke kiri/ke kanan :

Simbol : 

Ke atas/ ke bawah :

Simbol : 

### 5. Terminal

Menunjukkan awal, akhir atau titik interupsi.

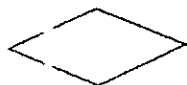
Simbol :



### 6. Testing

Memperlihatkan kondisi yang harus dites.

Simbol :



Contoh penggunaan flowchart :

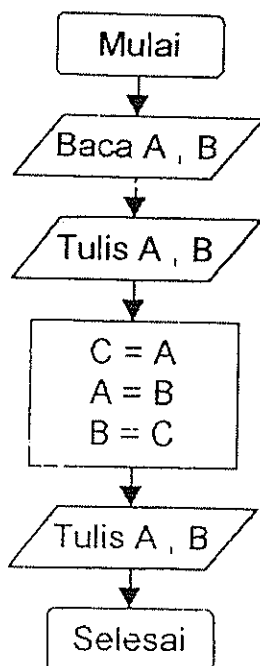
1. Bentuk algoritma dan flowchart untuk memindahkan nilai A dan B.

Penyelesaian :

Bentuk algoritma :

1. Mulai
2. Baca A , B
3. Tulis A , B
4.  $C = A$
5.  $A = B$
6.  $B = C$
7. Tulis A , B
8. Selesai

Bentuk flowchart :



2. Bentuk algoritma dan flowchart bila diketahui 10 buah data :  $X_1, X_2, \dots$

$X_{10}$  akan dihitung jumlah data tersebut.

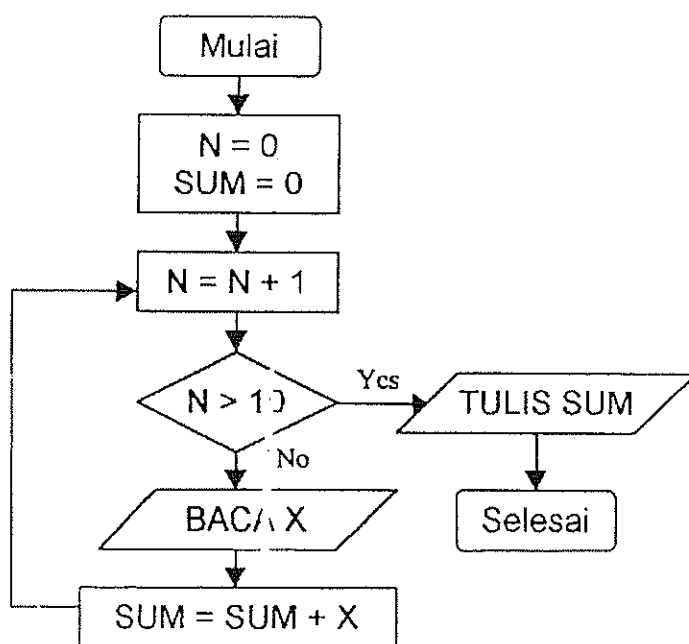
Penyelesaian :

Analisa :

Contoh data : 40, 25, -13, 19, 80, -27, 7, 64, 15, 30

N	X	SUM
1	40	40
2	25	65
3	-13	52
4	19	71
5	80	151
6	-27	124
7	7	131
8	64	195
9	15	210
10	30	240

Bentuk flowchart :



3. Bentuk algorithma dan flowchart bila diketahui N buah data :  $X_1, X_2, \dots, X_N$  akan dihitung jumlah data dan rata-rata dari data tersebut.

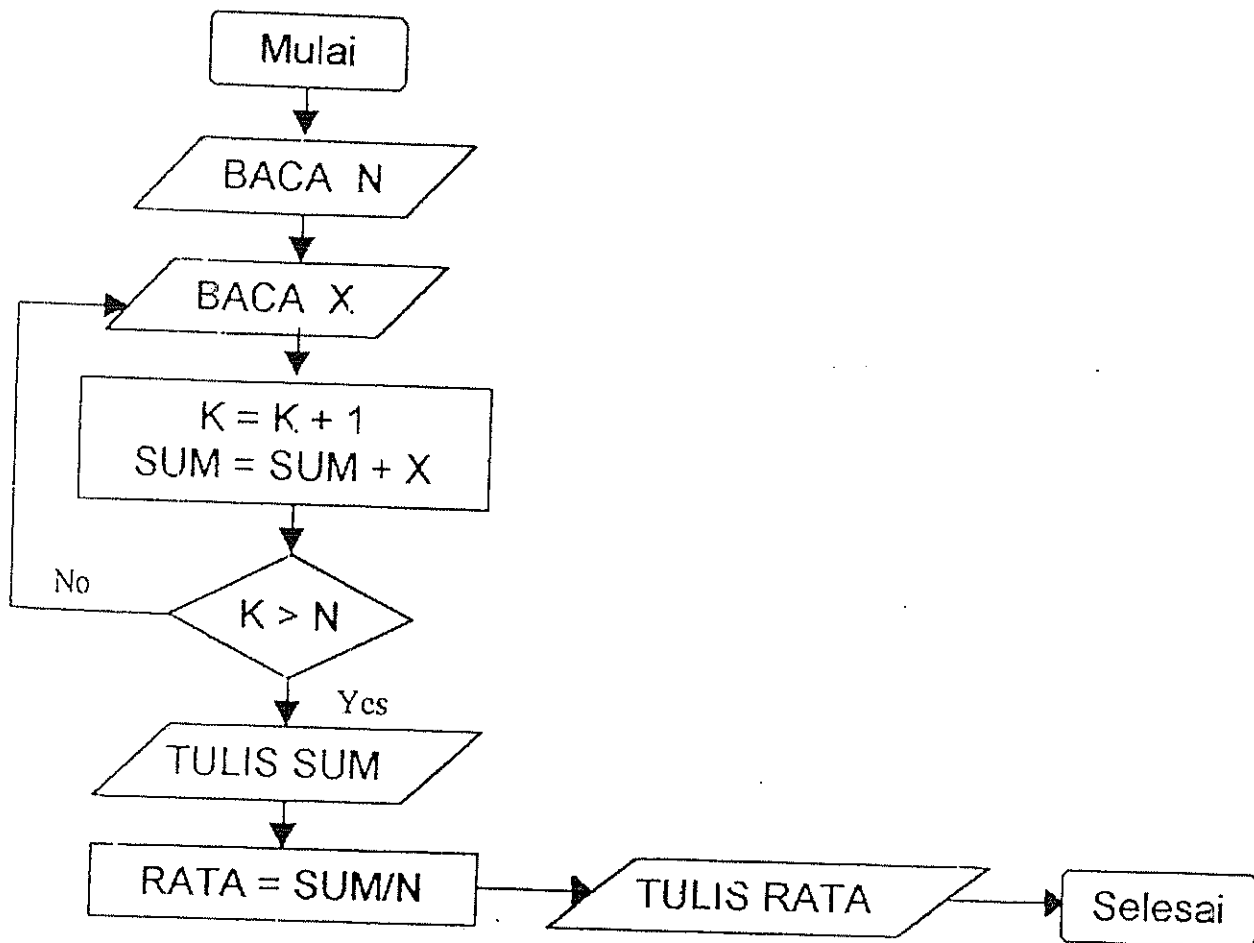
Penyelesaian :

Analisa :

Contoh data :  $N = 5$ , data : 10, 18, 29, 46, 22

N	K	X	SUM	RATA
5	1	10	10	25
	2	18	28	
	3	29	57	
	4	46	103	
	5	22	125	

Bentuk flowchart :



Buat Flowchart Program Berikut

1. Program mencari akar persamaan kuadrat

$$Y = AX^2 + BX + C$$

2. Program Hitungan Azimut (4 Kuadran) dari dua Koordinat (X1,Y1) dan (X2,Y2).

## BASIC

→ Beginner All Purpose Symbolic Instruction Code (Kode simbolik yang bersifat serbaguna bagi pemula)

Program Hitungan cocok digunakan untuk menghitung suatu perhitungan yang besar dan memerlukan waktu yang singkat.

Syarat – Syarat pemrograman

1. Menguasai permasalahan yang akan dikerjakan
2. Dapat menguasai komputer khususnya bahasa pemrograman

## DATA

→ penggambaran fakta secara umum yang dilukiskan dengan simbol tulisan atau dengan angka.

→ contoh : nama, jarak, sudut, tinggi, azimuth dll

Macam Data :

1. Data Numeris → data yang hanya dapat dinyatakan dengan angka, contoh jarak  $D1 = 250$  meter, sudut  $A1 = 35^{\circ}25'00''$
2. Data Alphanumeric/string → Data yang dapat dinyatakan dengan kombinasi angka & huruf, contoh : nama BM = BM 01, No. MHS = L2M0058

## KONSTANTA

→ Data yang besarnya tetap dan tidak berubah

## VARIABEL

→ Alat untuk menyimpan suatu konstanta



→ Dibagi 2 Variabel Numeris & Variabel Alphanumeric

→ Nama Variabel harus dimulai dengan huruf (A sampai Z) dan bisa diikuti huruf lain atau angka

Contoh :

$$\underline{5} * \underline{X}^2 + \underline{3} * \underline{X} * \underline{Y1} + \underline{6} * \underline{Y2}$$

K V K V V K V

Variabel Numeris dibedakan menjadi 3 jenis

1. Variabel numeris integer → variabel numeris yang harganya bulat, nama variabel ini ditambah dengan simbol % (persen), contoh : X%, A1%
2. Variabel numeris presisi tunggal (single precision) → Variabel numeris yang harganya pecahan dengan jumlah digit tidak lebih dari 7 digit, nama variabel ditambah dengan simbol ! (tanda seru), contoh : X!, A1!
3. Variabel numeris presisi ganda (double precision) → Variabel numeris yang harganya pecahan dengan jumlah digit tidak lebih dari 17 digit, nama variabel ditambah dengan simbol # (tanda palang), contoh : X#, A1#

$$X\% = 4/3 = 1$$

$$X! = 4/3 = 1.3333333$$

$$X\# = 4/3 = 1.333333333333333333$$

Presisi Tunggal

1. Panjang digit yang dapat dipercaya (significant) adalah 7

2. Penulisan titik mengambang dengan huruf E, contoh :  
0.125E-3
3. Setiap data membutuhkan tempat dalam memori sebanyak 4 byte, contoh bilangan 1.2 perlu 4 byte, bilangan 25000 perlu 4 byte juga.

#### Presisi Ganda

1. Panjang digit yang dapat dipercaya (significant) adalah 17
2. Penulisan titik mengambang dengan huruf D, contoh :  
0.125D-3
3. Setiap data membutuhkan tempat dalam memori sebanyak 8 byte, contoh bilangan 1.2 perlu 8 byte, bilangan 25000 perlu 8 byte juga.

Untuk tipe integer tiap konstanta memerlukan tempat dalam memori sebanyak 2 byte.

Matriks A(25,10):

Banyak elemen =  $25 \times 10 = 250$

Memori yang diperlukan:

→ presisi tunggal =  $250 \times 4 = 1000$  byte

→ presisi ganda =  $250 \times 8 = 2000$  byte

→ integer =  $250 \times 2 = 500$  byte

Untuk konstanta alphanumericis tiap variabel string perlu 3 byte + cacah string dalam konstanta, contoh NamaDesa\$="Tembalang" memerlukan  $3 + 9 = 11$  byte

## OPERATOR ARITMATIKA

Dalam bahasa basic dikenal 5 macam operator aritmetika :

1.  $^$  untuk pangkat
2.  $*$  untuk perkalian
3.  $/$  untuk pembagian
4.  $+$  untuk penjumlahan
5.  $-$  untuk pengurangan

Hierarki operator tersebut paling tinggi  $^$  kemudian  $*$  &  $/$  kemudian  $+$  &  $-$  tingkatan paling rendah.

Prinsip operator aritmetika :

1. Operator yang berada di kiri akan dikerjakan lebih dahulu
2. Bila hierarkinya berlainan, maka yang hierarkinya lebih tinggi akan dikerjakan dahulu.
3. Operator yang terletak di dalam kurung dikerjakan lebih dahulu.
4. Bila ada beberapa kurung maka kurung yang terdalam dikerjakan lebih dahulu.
5. Variabel real/integer tidak boleh dibagi dengan 0
6. pangkat bilangan negatif tidak boleh

## OPERATOR HUBUNGAN

Operator	Operasi	Contoh
=	Kesamaan	$A = B$
$\neq$ atau $\neq$	Ketidaksamaan	$A \neq B$ atau $A \neq B$
<	Kurang (lebih kecil) dari	$A < B$
>	Lebih besar dari	$A > B$
$\leq$ atau $\leq$	Kurang dari atau sama dengan	$A \leq B$ atau $A \leq B$
$\geq$ atau $\geq$	Lebih besar atau sama dengan	$A \geq B$ atau $A \geq B$

## STATEMEN

→ Suatu rangkaian instruksi yang akan memerintahkan komputer untuk membentuk atau melakukan suatu operasi.

Dalam Bahasa BASIC statemen dapat digolongkan menjadi lima golongan yaitu :

1. Statemen Aritmetika, contoh : rem, let
2. Statemen Input/Output, contoh : read-data, input-print
3. Statemen Kendali, contoh : goto, if-then
4. Statemen Function/Subprogram, contoh : Sin (X), log (x)
5. Statemen spesifikasi, contoh : dimension, common

## FUNGSI – FUNGSI NUMERIK DALAM BASIC

### 1. Fungsi Trigonometri

SIN (X)	}	dalam radian
COS (X)		
TAN (X)		
ATN (X)		

2. EXP (X) → menyatakan e pangkat x

3. LOG (X) → menyatakan logaritma bilangan pokok e dari X

4. ABS (X) → menyatakan nilai absolut dari X

5. SQR (X) → menyatakan akar pangkat 2 dari X

6. INT (X) → menyatakan bilangan bulat terbesar yang tidak melebihi X

## Penulisan Rumus-rumus dengan operator Aritmetika

1.  $A = 4X^2Y - 3XY + 7YZ^3$

2.  $A = \left[ \frac{\frac{a^2}{b} + 6c^5}{x + \frac{y}{z}} \right]$

3.  $A = \frac{x^{n+1}}{y^{\frac{2n+3}{2}}}$

4.  $M = \frac{a(1 - e^2)}{(1 - e^2 \sin^2 Q)^{3/2}}$

## STATEMEN ARITMETIKA

### 1. Statemen REM

→ Statemen yang tidak perlu diproses dalam komputer, hanya merupakan suatu komentar

→ Bentuk Umum : NO REM KOMENTAR

→ Contoh :

10 REM PROGRAM MENHITUNG LUAS

20 REM DIBUAT OLEH

## 2. Statemen LET

→ Statemen untuk menyatakan suatu operasi aritmatika

→ Bentuk Umum : NO LET VAR = ARGUMEN

→ Contoh :

10 LET P = 5

20 LET L = 2

30 LET LUAS = P\*L

Kesalahan-kesalahan yang sering terjadi dalam penulisan ekspresi :

a. 10 LET 4+B = A →

b. D10 LET A = 20 →

c. 10 LET C = (A+5)(B+10) →

d. 10 L = SQR (S\*(S-A)\*(S-B)\*(S-C)

e. 10 2L = 2\*P\*L

f. 10 L = P+\*L

g. 10 L = 1,200

h. 10 L = 7.5-



## STATEMENT INPUT – OUTPUT

### I. INPUT

#### 1. Statemen READ dan Deklarasi DATA

→ Statemen READ digunakan untuk memberikan harga pada variabel dengan membaca harga tersebut pada deklarasi DATA

→ Bentuk Umum :

NO READ VAR1,VAR2,VAR3,....

NO DATA HARGA1, HARGA2, HARGA3,....

→ Contoh :

10 REM PROGRAM HITUNG LUAS

20 READ P,L

30 LET LU = P\*L

40 PRINT "LUAS=";LU

50 DATA 10,5

60 END

run

keluaran

LUAS= 50

→ Jumlah konstanta deklarasi DATA harus sama dengan jumlah variabel dalam statemen READ. Untuk menuliskan beberapa

nama variabel dalam satu statemen READ harus dipisahkan dengan tanda KOMA

→ Jika jumlah data kurang, komputer akan menyatakan data kurang, jika jumlah data lebih akan diabaikan

## 2. Statemen Input

→ Statemen input memasukkan data yang akan diolah oleh komputer saat eksekusi berlangsung

→ Bentuk Umum :

NO INPUT [STRING]

→ Contoh :

```
10 REM PROGRAM HITUNG LUAS
```

```
20 INPUT "PANJANG SISI =";P
```

```
30 INPUT "LEBAR SISI =";L
```

```
40 LET LU = P*L
```

```
50 PRINT "LUAS =";LU
```

```
60 END
```

run

keluarannya

```
PANJANG SISI =? 10
```

```
LEBAR SISI =? 5
```

```
LUAS = 50
```

## 3. Statemen RESTORE

→ Statemen RESTORE dapat menggerakkan program ke suatu konstanta yang dituju, artinya konstanta yang telah dibaca bias dibaca kembali

→ Bentuk Umum :

NO RESTORE NO

→ Contoh :

10 READ A,B,C,D

20 LET E = A+B+C+D

30 RESTORE 60

40 READ P,Q,R,S

50 LET T = P+Q+R+S

60 DATA 10,5,10,5

70 PRINT E

80 PRINT T

90 END

RUN

Keluarannya

30

30

II. OUTPUT

## 1. Statemen PRINT

→ Statemen PRINT untuk menampilkan informasi data hasil pengolahan komputer lewat layar monitor

→ Bentuk Umum :

NO PRINT "STRING",VAR1,VAR2,.....

## 2. Statemen LPRINT

→ Statemen LPRINT untuk menampilkan informasi data hasil pengolahan komputer lewat printer

→ Bentuk Umum :

NO LPRINT "STRING",VAR1,VAR2,.....

→ Contoh

10 REM PROGRAM HITUNG LUAS

20 INPUT "PANJANG SISI =";P

30 INPUT "LEBAR SISI =";L

40 LET LU = P\*L

50 PRINT "LUAS =";LU → Lewat layar monitor

60 LPRINT "LUAS =";LU → Lewat Printer

70 END

→ Output dapat dipisahkan dengan koma ( , ) atau dengan titik koma ( ; )

→ Jika dipisahkan dengan koma hasilnya akan dicetak pada zone yang berbeda

→ Jika dipisahkan dengan titik koma hasilnya akan dicetak secara berdampingan

→ Pembagian zone pada layar monitor komputer dipisahkan dengan tanda koma :

0	15	16	31	32	47	48	63	64
ZONE I		ZONE II		ZONE III		ZONE IV		

→ Contoh:

10 A = 10:B = 20:C = 30

20 PRINT A,B,C

30 PRINT A;B;C

40 END

RUN

10                      20                      30

10 20 30

Coba buat program sederhana berikut :

1. Program hitungan luas segitiga dengan data panjang ketiga sisinya.
2. Program hitungan jarak dua titik dari data koordinat titik tersebut
3. Program hitungan jarak optis & beda tinggi dengan data pengukuran theodolit

### 3. Statemen PRINT TAB

→ Statemen PRINT TAB untuk mengatur jarak dari satu item ke item lainnya

→ Bentuk Umum :

NO PRINT TAB (m);VAR1;TAB (n);VAR2...

m, n → bilangan bulat (dari 0 s/d 64) atau ekspresi untuk menunjukkan dimulai dari kolom berapa suatu karakter akan dicetak.

VAR1,VAR2 → item yang akan dicetak, yang berupa konstanta, ekspresi numeris atau string, variabel numeris atau literal string.

→ Contoh

```
10 REM PROGRAM HITUNG LUAS
20 INPUT "PANJANG SISI =";P
30 INPUT "LEBAR SISI =";L
40 LET LU = P*L
50 PRINT TAB (0);"PANJANG";TAB (10);"LEBAR";TAB
(20) "LUAS"
60 PRINT TAB (3);P;TAB(12);L;TAB(22);LU
70 END
```

#### 4. Statemen PRINT USING

→ Statemen PRINT USING untuk mengatur format output

→ Bentuk Umum :

i) NO PRINT USING "KAR";VAR1;VAR2....

ii) NO PRINT USING A\$;VAR1;VAR2

KAR = Karakter bayangan yang berfungsi sebagai tempat kedudukan karakter dari item yang akan dicetak.

Karakter bayangan

a. Karakter bayangan numeris

# → dipakai untuk menentukan digit dalam numeris

+, - → jika tanda ini ditempatkan di awal atau akhir suatu medan, maka tanda tersebut akan dicetak sesuai dengan spesifikasinya

\*\* → dua tanda bintang yang ditempatkan sebelah kiri medan akan mengakibatkan sisa ruangan di sebelah kiri hasil cetakkan diisi tanda "\*"

^^^ → tanda ini menyebabkan bilangan ditulis dalam format eksponensial dengan bentuk E+xx atau D+xx

→ Contoh

```
CLS
```

```
PRINT USING "##.##";0.78
```

```
PRINT USING "###.##";987.654
```



```

PRINT USING "##.##";1032.6789
PRINT USING "+##.### +###.##";23.34,-345.56
PRINT USING "##.###+ ###.##+";23.34,-345.56
PRINT USING "##.### -###.##";23.34,-345.56
PRINT USING "##.###- ###.##-";23.34,-345.56
PRINT USING "###.##";14.2,-6.4
PRINT USING "#.##^ ^ ^";234.56
END

```

RUN

0.78

987.65

%1032.68

+23.340 -345.56

23.340+ 345.56-

23.340 -345.56 ||

23.340 345.56-

14.2 -6.4

2.35E+02

b. Karakter bayangan string

! → akan tercetak karakter pertama dari string

& → semua karakter dari data string akan tercetak

\\ → cacah karakter yang akan tercetak

→ Contoh

```
CLS
```

```
A$="DASAR PEMROGRAMAN KOMPUTER"
```

```
PRINT USING "!";A$
```

```
PRINT USING "&";A$
```

```
PRINT USING "\ \";A$
```

```
END
```

```
RUN
```

```
D
```

```
DASAR PEMROGRAMAN KOMPUTER
```

```
DASAR
```

## STATEMENT KENDALI

### 1. Statemen GOTO

→ Statemen GOTO untuk melompat proses eksekusi sehingga komputer melakukan eksekusi tidak dengan urutan normal

→ Bentuk Umum :

NO GOTO NO BARIS YANG DITUJU

→ Contoh :

10 REM PROGRAM HITUNG LUAS

20 INPUT "PANJANG SISI =";P

30 INPUT "LEBAR SISI =";L

40 LU = P\*L

50 PRINT "LUAS=";LU

60 GOTO 20

70 END

RUN

PANJANG=?10

LEBAR=?5

LUAS=50

PANJANG=?20

LEBAR=?10

LUAS=100

PANJANG=?

## DAN SETERUSNYA

### 2. Statemen IF ... THEN ... ELSE

→ Fungsinya untuk melaksanakan pengetesan atas suatu nalar

→ Bentuk Umum :

```
NO IF EKSPRESI NALAR THEN STATEMEN 1 ELSE  
STATEMEN 2
```

STATEMEN NALAR = Pembandingan antara dua variabel atau salah satu berupa konstanta.

STATEMEN 1 = Akan dikerjakan jika ekspresi nalar dipenuhi

STATEMEN 2 = Akan dikerjakan jika ekspresi nalar tidak dipenuhi

→ Contoh :

```
10 INPUT "Nilai Mahasiswa=";N  
20 IF N>60 THEN PRINT "LULUS" ELSE PRINT "TIDAK  
LULUS"  
30 END  
RUN  
Nilai Mahasiswa=? 70  
LULUS
```

→ Contoh :

```
cls  
input "Nilai Mahasiswa=";n
```

```

if n>90 then NH$="A" else if n>80 then NH$="B" else if
    n>70 then NH$="C" else if n>60 then NH$="D" else
    NH$="E"

```

```

print

```

```

Print "Nilai Angka";tab(15) "Nilai Huruf"

```

```

print "=====

```

```

print tab(5)n;tab(20)NH$

```

```

end

```

```

run

```

Nilai Mahasiswa=? 75

```

Nilai Angka      Nilai Huruf

```

```

=====

```

```

75

```

```

C

```

### 3. Statemen FOR-NEXT

→ Fungsinya memberikan instruksi komputer untuk melakukan operasi berulang

→ Bentuk Umum :

```

NO FOR VAR1=EKS1 TO EKS2 STEP EKS3

```

```

....

```

```

....

```

```

NO NEXT VAR1

```

→ Contoh :

```
10 REM CONTOH PENGGUNAAN STATEMEN FOR-  
NEXT
```

```
15 K=5
```

```
20 FOR I=1 TO 5
```

```
25 M=K*I
```

```
30 PRINT "A(";I;")=";M
```

```
40 NEXT I
```

```
50 END
```

run

A( 1 )= 5

A( 2 )= 10

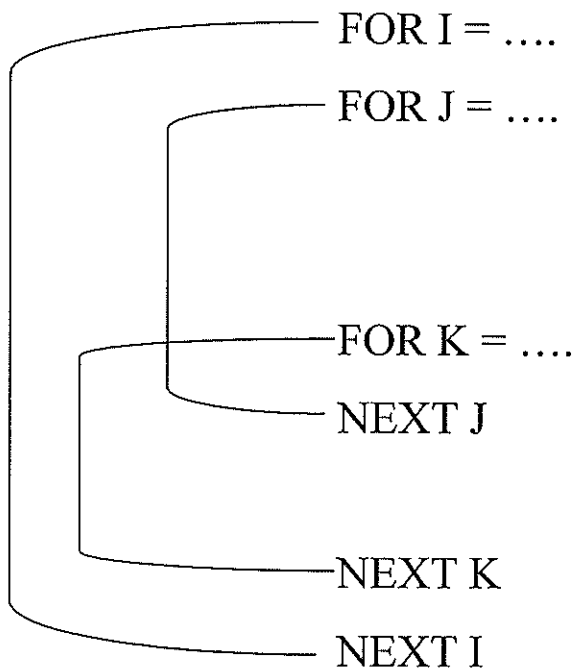
A( 3 )= 15

A( 4 )= 20

A( 5 )= 25

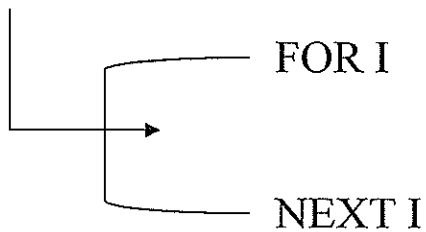
Syarat yang tidak diperbolehkan dalam kalang :

1. Operasi kalang tidak boleh saling berpotongan



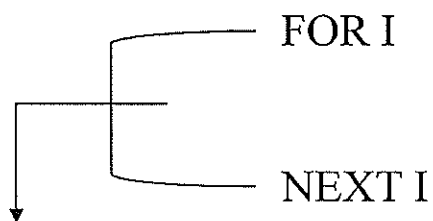
2. Eksekusi loncat ke dalam kalang tidak diperbolehkan

Statemen Masuk



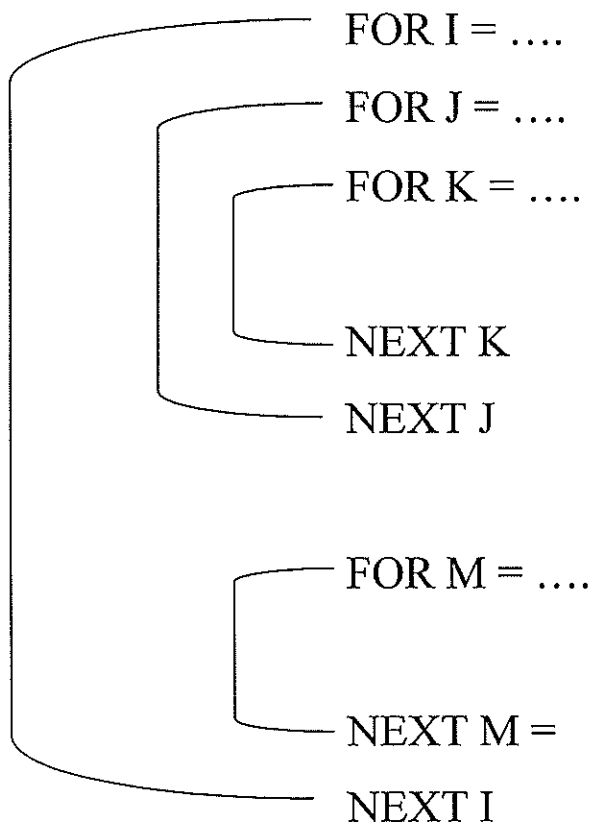
Syarat yang diperbolehkan dalam operasi kalang

1. Eksekusi loncat keluar dari kalang diperbolehkan



Statemen Keluar

2. Operasi dalam kalang diperbolehkan



#### 4. Statemen IF NALAR GOTO NO

→ Bentuk Umum :

NO IF NALAR GOTO NO

→ Contoh :

10 CLS

20 INPUT "NAMA MAHASISWA=";M\$

30 INPUT "NILAI UJIAN=";N

40 PRINT:PRINT "NAMA MAHASIWA";TAB(20)"NILAI  
UJIAN";TAB(35)"HASIL UJIAN"

50 PRINT

"=====

60 IF N>=60 GOTO 90



```
70 H$="TIDAK LULUS"  
80 GOTO 100  
90 H$="LULUS"  
100 PRINT M$;TAB(23)N;TAB(35)H$  
110 END
```

#### 5. Statemen IF NALAR THEN OPERASI ARITMATIKA

→ Bentuk Umum :

NO IF NALAR THEN OPERASI ARITMATIKA

→ Contoh :

```
10 CLS  
20 INPUT "A=";A  
35 INPUT "B=";B  
40 C=A+B  
50 IF C>=15 THEN C=15  
60 PRINT:PRINT "C=";C  
70 END
```

#### 6. Statemen IF NALAR THEN NO

→ Bentuk Umum :

NO IF NALAR THEN NO

→ Contoh :

```
10 CLS  
20 INPUT "NAMA MAHASISWA=";M$
```

```
30 INPUT "NILAI UJIAN=";N
40 PRINT:PRINT "NAMA MAHASIWA";TAB(20)"NILAI
   UJIAN";TAB(35)"HASIL UJIAN"
50                                     PRINT
   "=====
60 IF N>=60 THEN 90
70 H$="TIDAK LULUS"
80 GOTO 100
90 H$="LULUS"
100 PRINT M$;TAB(23)N;TAB(35)H$
110 END
```

## SUBROUTIN

- Merupakan bagian dari seluruh program (subprogram).
- Operasi dalam subrutin ini hanya mengambil harga dari variabel-variabel yang sebelumnya sudah diketahui, sehingga untuk melaksanakan operasi tersebut seolah-olah dapat berdiri sendiri

### Fungsi Subroutine

1. Dapat dipakai untuk melakukan operasi yang sama berulang kali
2. Menghemat memori dan waktu
3. Menjadikan program lebih sederhana efisien tetapi dapat menyelesaikan persoalan yang cukup rumit
4. Untuk melakukan suatu operasi yang sama komputer cukup memanggil subrutin tersebut.

### 1. STATEMEN GOSUB

→ Bentuk Umum :

NO GOSUB NO BARIS YANG DITUJU

....

....

NO BARIS YANG DITUJU

...

...

## NO RETURN

Statemen subroutine harus diakhiri dengan statemen RETURN

→ Contoh :

```
10 REM PROGRAM HITUNGAN AZIMUT AB DAN BA
20 REM DIBUAT OLEH TEKNIK GEODESI FT UNDIP
30 CLS
40 INPUT "XA =";XA:INPUT "YA =";YA:
   INPUT "XB =";XB:INPUT "YB =";YB
50 PI=4*ATN(1)
60 REM HITUNGAN AZIMUT AB
70 DX=XB-XA:DY=YB-YA
80 ARAB=ATN(DX/DY):ADAB=ARAB*180/PI
90 A=ADAB
100 GOSUB 300
110 ADGAB=B
120 REM HITUNGAN AZIMUT BA
130 DX=XA-XB:DY=YA-YB
140 ARBA=ATN(DX/DY):ADBA=ARBA*180/PI
150 A=ADBA
160 GOSUB 300
170 ADGBA=B
180 PRINT
190 PRINT "KOORDINAT A";TAB(15)"KOORDINAT B";
   TAB(30)"AZIMUT AB";TAB(45)"AZIMUT BA"
200 PRINT
"====="
```

```
210                                                    PRINT
"(";XA;",";YA;");TAB(15)(";XB;",";YB;");TAB(30)
    ADGAB;TAB(45)ADGBA
220 END
300 REM SUBROUTINE AZIMUT
310 IF DX>0 AND DY>0 THEN B=A ELSE IF DX>0 AND
DY<0
    THEN B=A+180 ELSE IF DX<0 AND DY<0 THEN
B=A+180
    ELSE B=A+360
320 RETURN
```

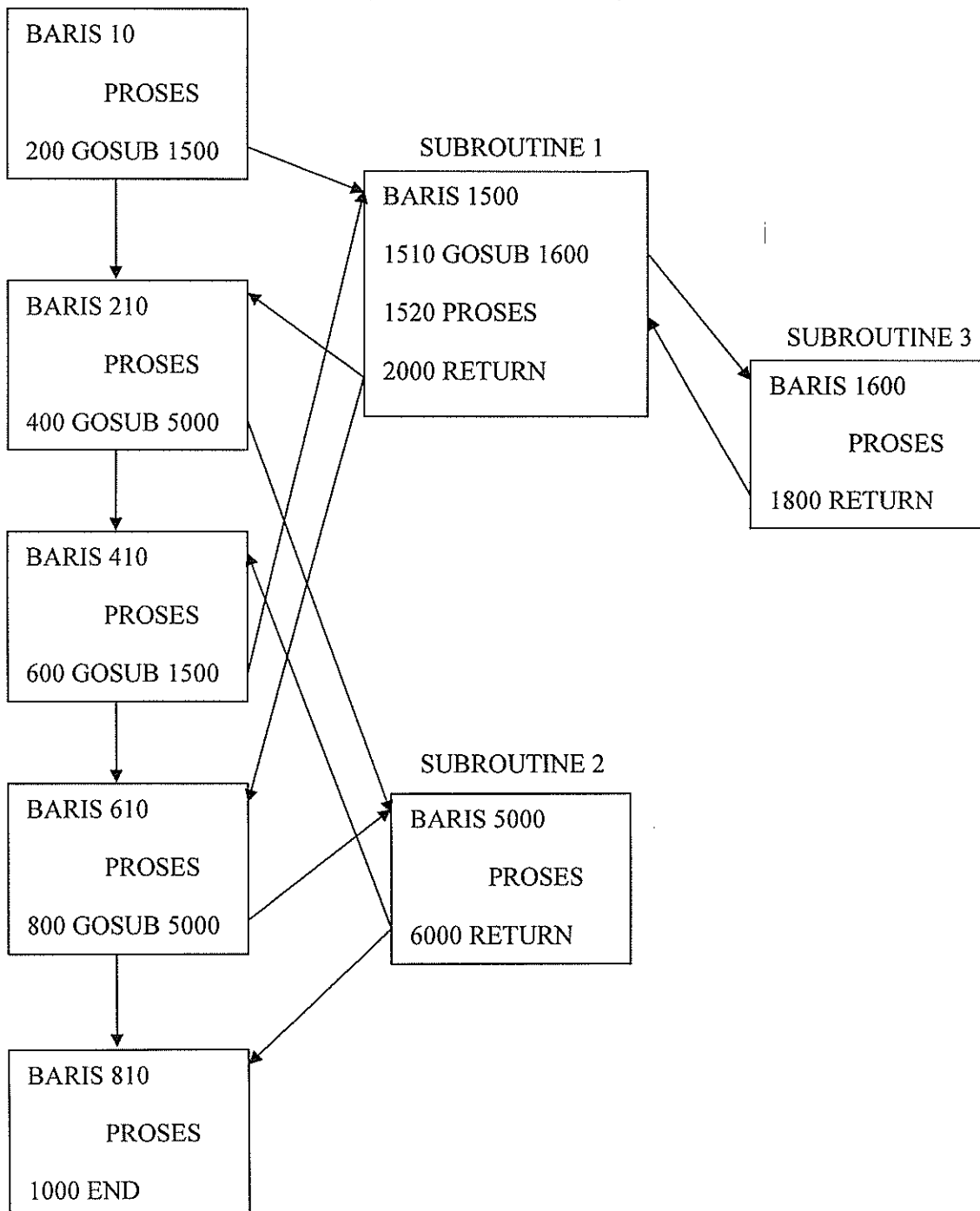
## 2. STATEMEN RETURN CLEAR

→ Fungsinya untuk menghapus subroutine yang ada di dalam program supaya memorinya dapat dihemat

→ Bentuk Umum :

### NO RETURN CLEAR

Dalam Program Utama diperbolehkan adanya subroutine lebih dari satu :



Subroutine Dalam → subroutine yang dipanggil pada saat sebuah program utama dikerjakan

Subroutine Luar → subroutine yang dipanggil pada saat sebuah subroutine dikerjakan

### 3. STATEMEN DEF FN

→ Fungsinya hampir seperti subroutine yaitu berupa fungsi yang dapat digunakan berkali - kali

→ Bentuk Umum :

NO DEF FN VAR1(ARG1,ARG2,...) = EKSPRESI FUNGSI

→ Contoh :

## STATEMEN SPESIFIKASI

→ digunakan untuk membedakan variabel yang berindeks dengan variabel biasa.

### 1. STATEMEN DIM

→ Fungsinya untuk mendefinisikan variabel berindeks atau matrks

→ Bentuk Umum :

NO DIM VAR (KONSTANTA)

atau

NO DIM VAR\$ (KONSTANTA)

dimana :

VAR : Variabel numeris yang akan didefinisikan

VAR\$ : Variabel string/alphanumeric yang akan didefinisikan

KONSTANTA : Bilangan bulat

→ Contoh :

DIM A(100), B(200), A\$(50), B\$(10)

PROGRAM DIM1.BAS, DIM2, DIM3, DIM31,DIM312,DIM\$

SORTIR,SORTIR\$



## MATRIKS

→ disebut pula larik berdimensi dua, yaitu posisi setiap item data ditentukan oleh dua atau lebih ekspresi. Ekspresi tersebut disebut baris dan kolom.

→ Bentuk Umum :

NO DIM VAR (KONSTANTA1, KONSTANTA2)

atau

NO DIM VAR\$ (KONSTANTA1, KONSTANTA2)

dimana :

VAR : Variabel numeris yang akan didefinisikan

VAR\$ : Variabel string/alphanumeric yang akan didefinisikan

KONSTANTA : Bilangan bulat

→ Contoh :

DIM A(3,2), B(5,5), A\$(4,2), B\$(10,5)